



Del 3 – Sensorer

Avancerad elektronik och programmering

I den här labben fortsätter vi att bygga på de kunskaper vi fått i tidigare laborationer gällande programmering och elektronik. När man deltar i en CanSat-tävling ska man utföra några uppdrag och framförallt ska man åtminstone ha ett primärt uppdrag (eng. primary mission) där man mäter temperatur och lufttryck. I den här labben får du lära dig att använda en sensor som gör just detta, samt ytterligare programmeringskunskaper för att underlätta arbetet.

Material

- Adafruit feather RP2040
- Kopplingsdäck
- USB-sladd för att koppla feather till datorn
- Dator
- BMP180 breakout
- Sladdar för koppling

Sensorer och kommunikation

Både BMP180 (trycksensor) och MPU6050 (accelerometer och gyro) kommunicerar med Feather-kortet via en protokollstandard som kallas I2C (Inter-Integrated Circuit). I2C gör det möjligt för sensorer att kommunicera med Feather genom bara två huvudanslutningar, en för data och en för klocka:

- **SDA (data):** För att skicka och ta emot data.
- **SCL (klocka):** För att synkronisera kommunikationen mellan enheterna.

Anslutningar på Feather-kortet:

- SDA-pinnen på Feather ska kopplas till SDA på sensorn.
- SCL-pinnen på Feather ska kopplas till SCL på sensorn.

E-post: cansat@au.se

Telefon: 070-000 90 56

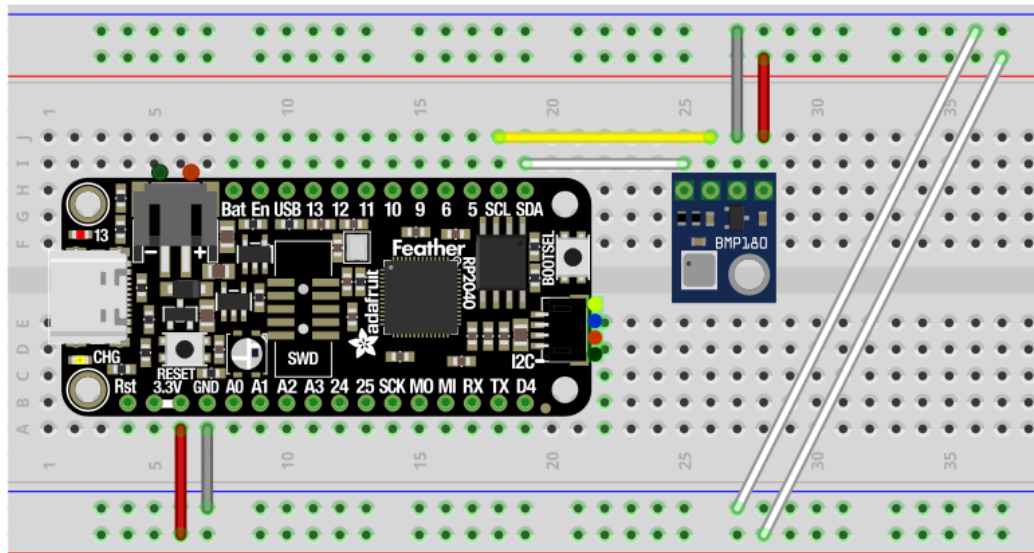
Senast uppdaterad: 2024-11-19



**Astronomisk
Ungdom**


Även ström behöver tillföras till sensorerna, detta görs genom att koppla in deras jordanslutning i form av GND till GND på feathern, samt deras positiva spänning, kallat VIN eller VCC till 3,3V på feathern.

Notera: Både BMP180 och MPU6050 är I2C-kompatibla och kan anslutas till samma SDA- och SCL-pinnar på Feather samtidigt.



Kod för att kommunicera med sensorn

För att läsa av trycksensorns (bmp180) mätningar kan följande kod användas:



```
code.py
1 import time
2 import board
3 import bmp180
4
5 # Ställ in BMP180-sensorn
6 i2c = board.I2C() # Initierar I2C-kommunikation
7 bmp = bmp180.BMP180(i2c)
8
9 # Justera denna konstant om nödvändigt för korrekt höjdberäkning
10 bmp.sea_level_pressure = 1013.25 # Standard lufttryck vid havsnivå (hPa)
11
12 while True:
13     # Läs och skriv ut temperatur och lufttryck
14     print("Temperatur:", bmp.temperature, "C")
15     print("Lufttryck:", bmp.pressure, "hPa")
16     print("Höjd:", bmp.altitude, "m")
17     time.sleep(1)
18
```

Extra utmaning

Koppla upp acceleration sensorn och följ följande exempelkod för att läsa dess data:

<https://docs.circuitpython.org/projects/mpu6050/en/latest/examples.html>

Biblioteket finns i Adafruit biblioteksamling och heter där adafruit_mpu6050.py

Testa att vicka runt på den lätt och se att värdena ändras.